Kharagpur Data Science Hackathon

ZELTA LABS

Problem Statement:

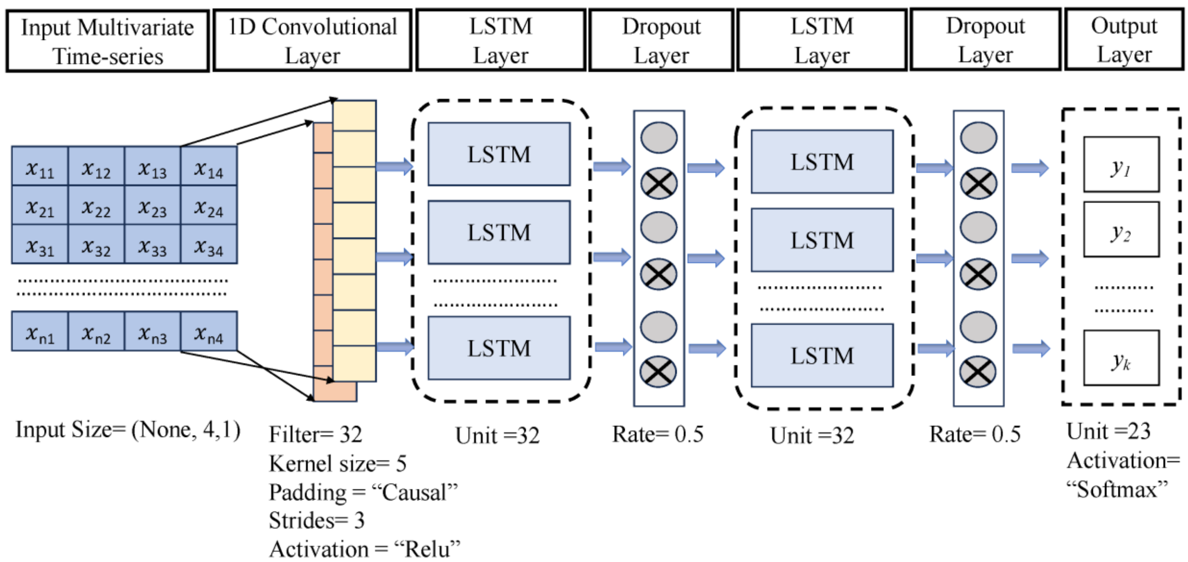
Algorithmic Trading Model Development for BTC/USDT Crypto Market

Our Approach :

To create a trading model for BTC/USDT with LSTM (Long short term memory), first we, gather and clean historical data, focusing on prices and volumes. Then we splitted the data for training and testing, then design the LSTM model. Train and assess the model, using it to predict future prices. Develop a trading strategy based on these predictions, backtest it, and refine the model and strategy iteratively. Integrate risk management measures like stop-loss orders, and deploy the model for live trading, keeping a close eye on its performance.

LSTM (Long short term memory)

Long Short-Term Memory (LSTM), a type of recurrent neural network (RNN), has found significant applications in finance due to its ability to capture and analyze sequential data.



In finance, where historical patterns often play a crucial role, LSTM excels in modeling and predicting time series data. Unlike traditional models, LSTMs can capture long-term dependencies and relationships within financial time series, making them particularly effective for tasks such as stock price prediction, risk management, and algorithmic trading. By understanding and remembering patterns in historical data, LSTMs enable more accurate predictions of future market trends. Their capacity to process and learn from sequential data has made them a valuable tool for financial analysts and traders seeking to leverage historical information for improved decision-making in dynamic and ever-changing markets.

Model Design and Architecture :

In designing our sequential LSTM model for predicting financial time series data, we carefully considered both accuracy and efficiency. The chosen architecture consists of multiple layers to effectively capture the sequential dependencies within the data.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 60, 50)            10400

 lstm_1 (LSTM)               (None, 64)                29440

 dense (Dense)               (None, 32)                2080

 dense_1 (Dense)             (None, 16)                528

 dense_2 (Dense)             (None, 1)                 17

=================================================================
Total params: 42465 (165.88 KB)
Trainable params: 42465 (165.88 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

The model begins with two LSTM (Long Short-Term Memory) layers, the first with 50 units and the second with 64 units, allowing the network to grasp both short-term and long-term patterns in the sequential data (understanding historical trends). Following the LSTM layers, we have introduced three dense layers. The first dense layer has 32 units, the second has 16 units, and the final layer produces a single output. These dense layers help in transforming the information learned by the LSTM layers into a format suitable for predicting the target variable.

The chosen architecture strikes a balance between complexity and computational efficiency, enabling effective learning from historical data while remaining practical for real-time prediction tasks in the dynamic financial markets.

Model Optimizing:

Here are some approaches to optimize the model.

1. **Hyperparameter Tuning:**
    - Adjust hyperparameters such as learning rate, batch size, and the number of LSTM units to find the optimal configuration.
2. **Regularization:**
    - Apply regularization techniques like dropout to prevent overfitting.
3. **Batch Normalization:**
    - Batch norm layers to normalize the inputs of each layer, which can accelerate training and improve model generalization.
4. **Learning Rate Scheduling:**
    - Techniques like step decay or exponential decay can help converge faster and avoid overshooting.
5. **Ensemble Methods:**
    - Combining predictions from multiple models, to improve overall performance. This could involve training multiple LSTM models with different initializations or architectures.

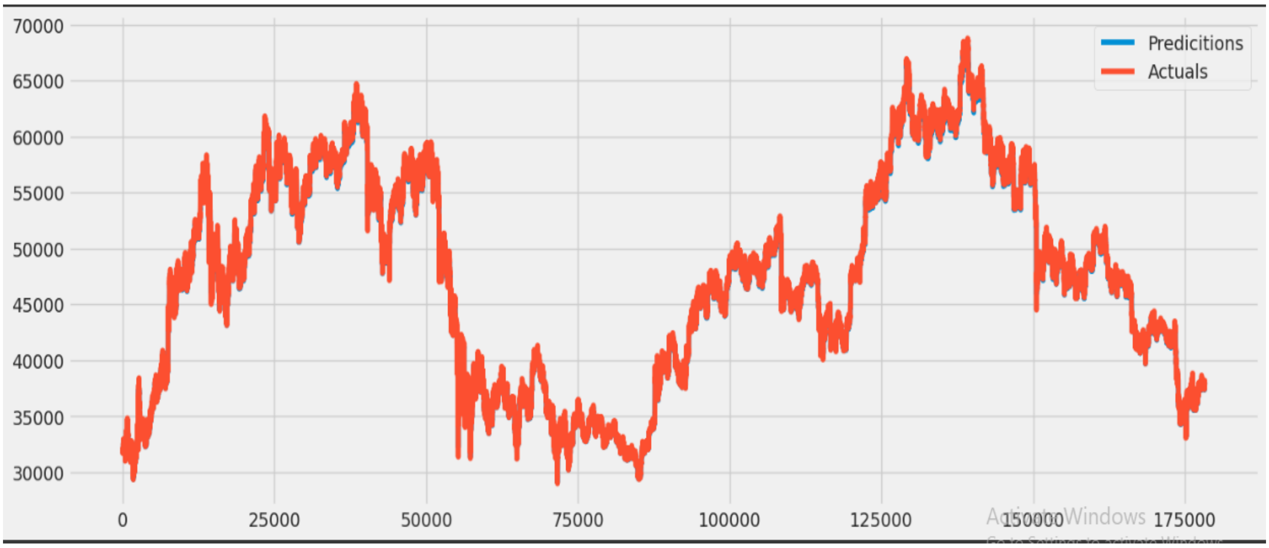
Also implementing these methods, we must also pay attention to Training set error, Dev set error and Test set error.

If the difference between The Human-level error as a proxy (estimate) for Bayes optimal error and Training set error is huge, there is huge avoidable bias which can be reduced by training a bigger neural network or optimizing algorithms (L2 optimization and dropout) or find any better neural network architecture .

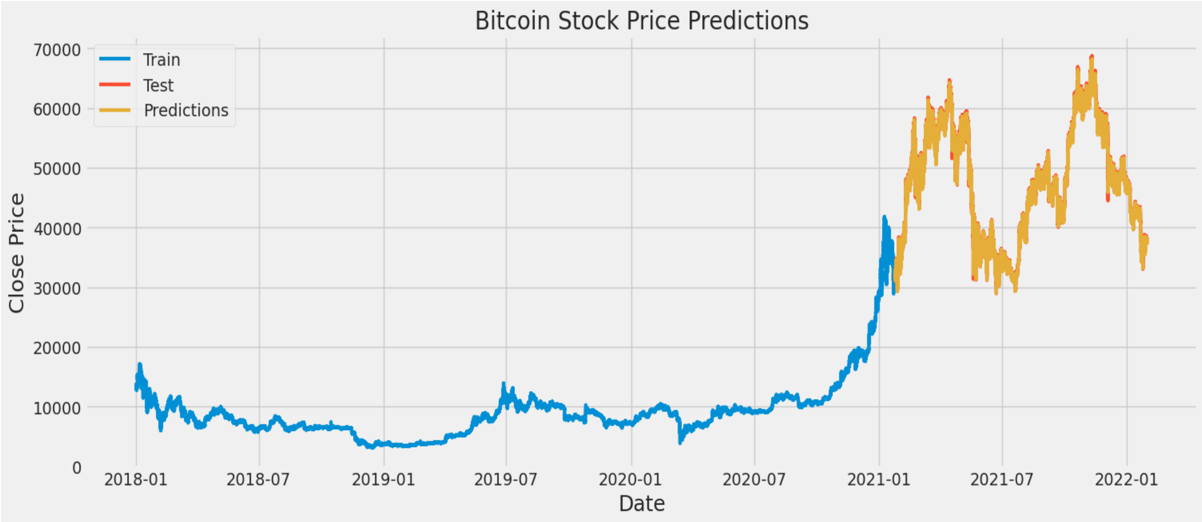
If the difference between Training set error and Dev set error is huge, there is a variance problem. To reduce variance problem we can increase the training set examples, regularization or find any better neural network architecture.

Backtesting Results:

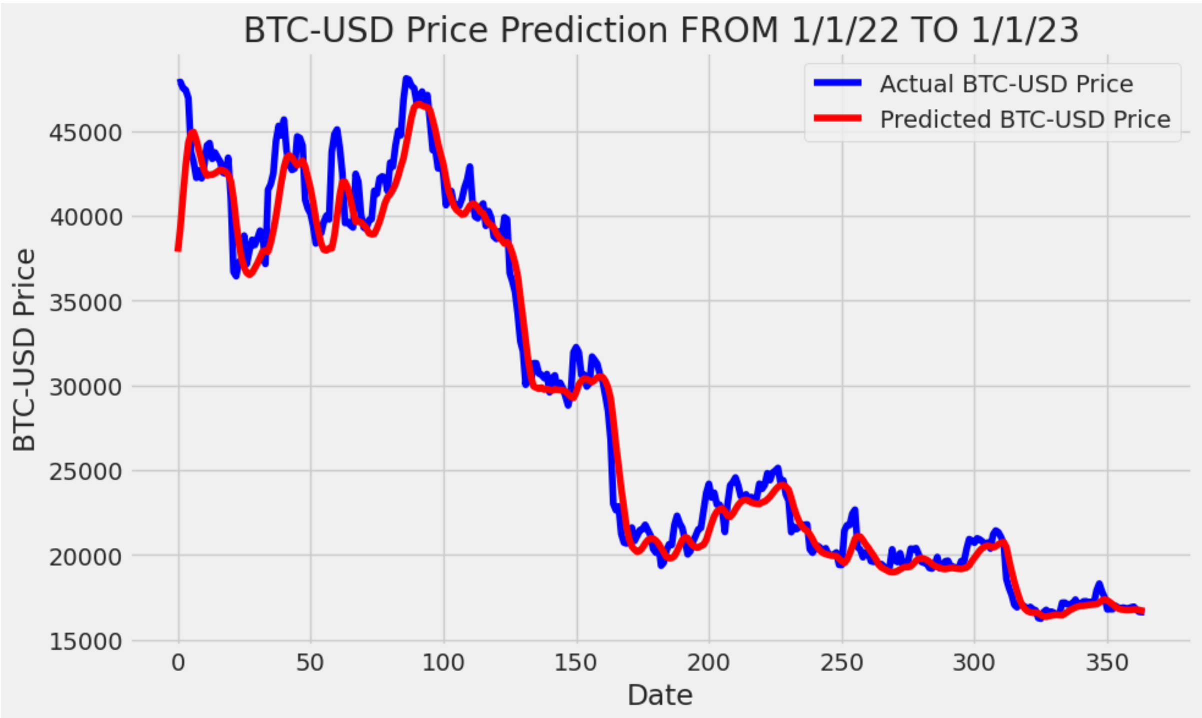
Backtesting is a crucial step in evaluating the performance of a predictive model using historical data. It involves applying a model to past data to simulate how it would have performed in real-world scenarios.



(Evaluating performance of model on test set)



(BTC Stock Price Predictions)



(Evaluating Performance of the model by comparing the prediction of model with actual stock values)

Parameters:

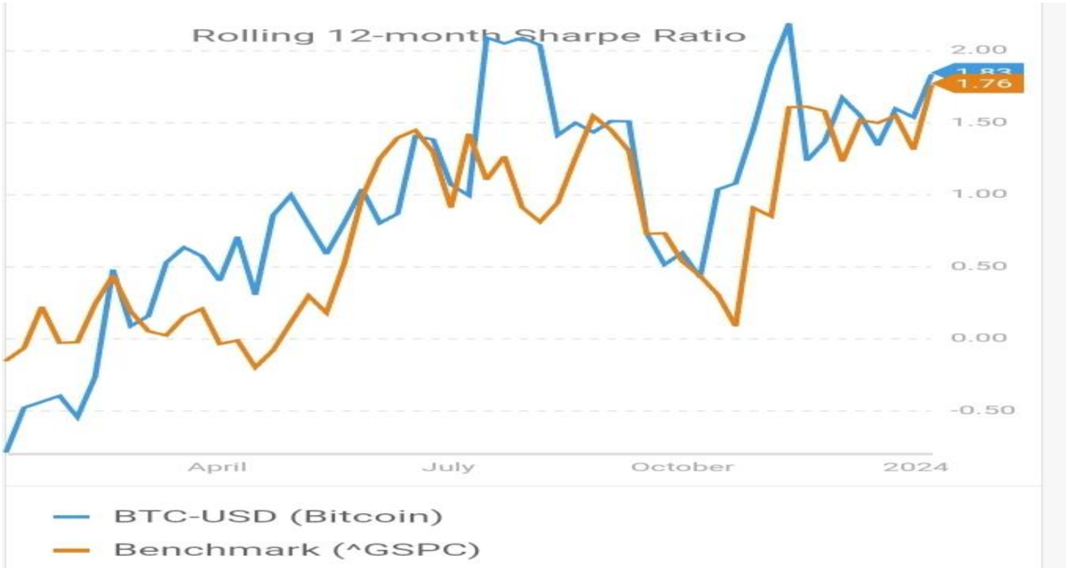
1. **Sharpe Ratio:**

- The Sharpe ratio is used to value the risk-adjusted return of an asset.

$$SharpeRatio = \frac{(Rp - Rf)}{\sigma_p}$$

- Rp = return of portfolio
- Rf = risk-free rate
- σ_p = standard deviation of the portfolio's excess return

- The current Bitcoin Sharpe ratio is **1.83**
- A Sharpe ratio **greater than 1.0** is considered acceptable.
- Following is a rolling 12 month Sharpe Ratio.



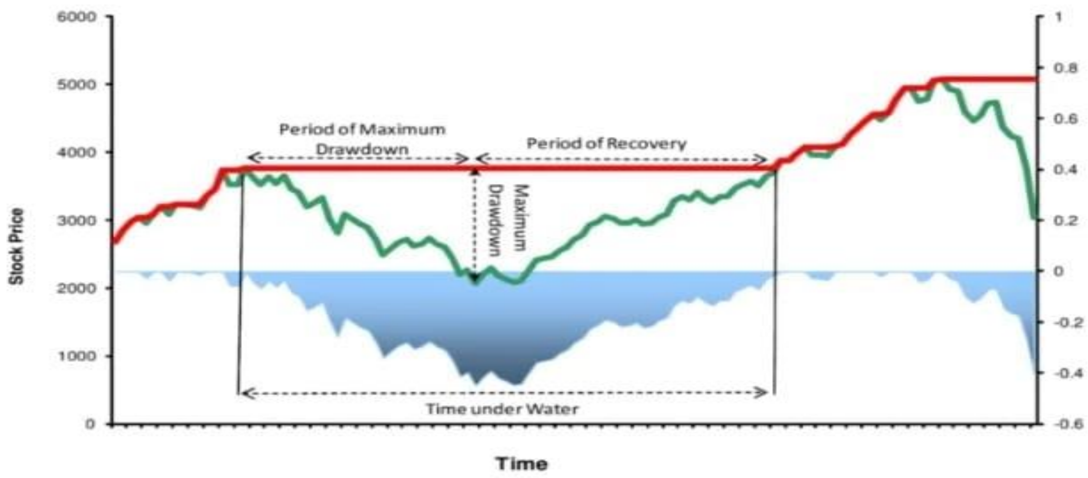
2. Annualized Returns:

- In the last 10 years, BTC obtained a 49.92% compound annual return, with a 77.36% standard deviation.
- Bitcoin Quarterly returns from 2013- present (%)

Time	Q1	Q2	Q3	Q4
2024	+9.26%			
2023	+71.77%	+7.19%	-11.54%	+56.9%
2022	-1.46%	-56.2%	-2.57%	-14.75%
2021	+103.17%	-40.36%	+25.01%	+5.45%
2020	-10.83%	+42.33%	+17.97%	+168.02%
2019	+8.74%	+159.36%	-22.86%	-13.54%
2018	-49.7%	-7.71%	+3.61%	-42.16%
2017	+11.89%	+123.86%	+80.41%	+215.07%
2016	-3.06%	+62.06%	-9.41%	+58.17%
2015	-24.14%	+7.57%	-10.05%	+81.24%
2014	-37.42%	+40.43%	-39.74%	-16.7%
2013	+539.96%	-3.97%	+40.6%	+479.59%
Average	+51.51%	+30.41%	+6.49%	+88.84%
Median	+3.64%	+7.57%	-2.57%	+56.90%

3. Maximum Drawdown:

- While MDD reflects past volatility, predicting future BTC price movements solely on MDD is unreliable.
- Market dynamics shift, making historical drawdowns not a definitive indicator of future dips.
- MDD in BTC context helps gauge downside risk. High MDD suggests potentially larger drops during corrections, guiding investors towards appropriate risk management strategies.



4. **Equity Curve**

- An equity curve is a graphical representation of the change in the value of a trading account over a time period.
- An equity curve with a consistently positive slope typically indicates that the trading strategies of the account are profitable, while a negative slope shows that they are generating a negative return.

Parameters calculated by performance of model :

Considering :

Invested Amount : $100,000

Slippage : 0.0015

Transaction Cost : 0.0015

Output :

Gross Profit: 672800.0827940726

Net Profit: 218604.4154755859

Total Closed Trades: 729

Win Rate (Profitability %): 43.895747599451305

Max Drawdown: 0.7586850511757554

Gross Loss: 702489.8845992682

Average Winning Trade (in USDT): 2102.5002587314766

Average Losing Trade (in USDT): 1717.579179949311

Buy and Hold Return of BTC: -11.369233370336925

Largest Losing Trade (in USDT): -11697.806610007297

Largest Winning Trade (in USDT): 15974.726027617198

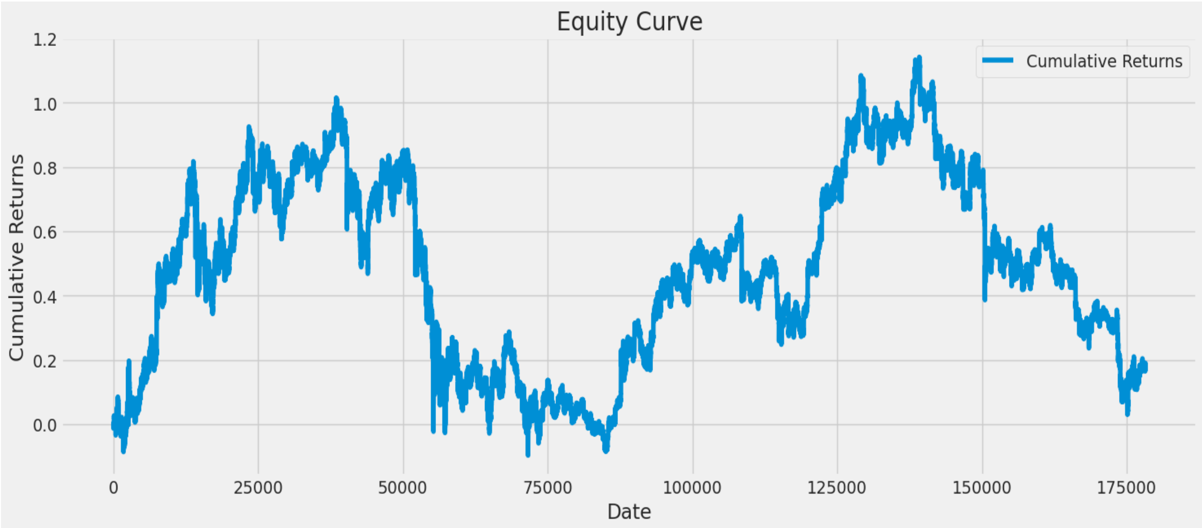
Sharpe Ratio: 0.014226571563827844

Sortino Ratio: 0.024614365120412453

Average Holding Duration per Trade: 0.9986301369863013

Max Dip in Running Trade: -7.0248988459926744

Average Dip in Running Trade: -4.047788438496201



(Equity Curve)

Risk Management :

In trading, it's crucial to manage risks effectively. This is a concise plan to help safeguard your capital and navigate market uncertainties. Various risk management mechanisms, such as stop-loss orders and hedging strategies, play a critical role in safeguarding capital in the BTC/USDT market. Evaluating their effectiveness is essential for investor confidence. Certain key aspects to consider in evaluating the risk management mechanisms are:

Maximum Drawdown Analysis	Measure the maximum drawdown during different market conditions. Analyze how well the risk management mechanisms limit the magnitude of losses during significant market downturns.

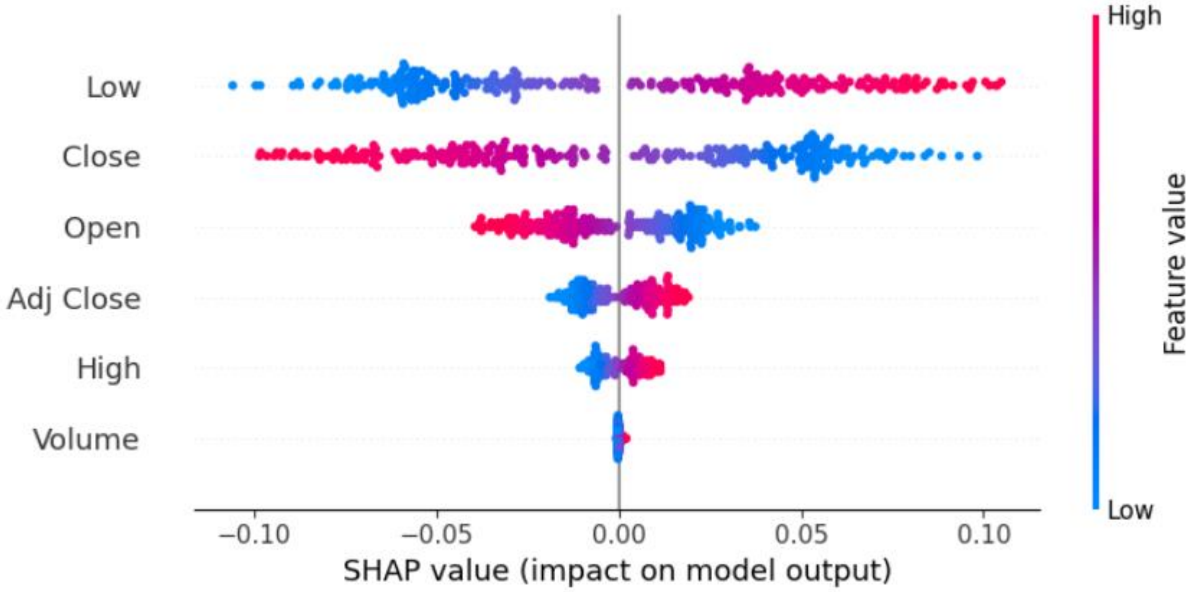
Risk-Adjusted Returns	Calculate risk-adjusted returns, such as the Sharpe ratio or Sortino ratio. These ratios consider the level of risk taken to achieve returns, providing a more nuanced evaluation of performance.
Volatility Control	Evaluate how well the risk management mechanisms control exposure to market volatility. Assess if the mechanisms adjust position sizes or allocation based on changes in market volatility.
Transaction Cost Analysis	Consider transaction costs associated with trading, including fees and slippage. Evaluate how well the risk management mechanisms account for these costs and whether they impact overall profitability.
Diversification Benefits	Assess the impact of diversification strategies on risk reduction. Determine if the risk management mechanisms effectively spread capital across different assets or strategies, reducing reliance on the performance of a single asset.
Position Sizing	Examine the impact of position sizing rules on capital preservation. Evaluate whether the risk management mechanisms appropriately adjust position sizes based on market conditions, asset volatility, or other relevant factors.
Stress Testing	Conduct stress tests to simulate extreme market scenarios. Assess how the risk management mechanisms perform under adverse conditions and whether they prevent catastrophic losses.
Adaptability to Market Conditions	Evaluate how well the risk management mechanisms adapt to changing market conditions. Assess whether the mechanisms are flexible enough to handle different phases of the market cycle.
Analysis of Risk Tolerance	Assess the alignment of risk management strategies with the investor's risk tolerance. Evaluate whether the mechanisms strike a balance between risk mitigation and the pursuit of returns.

Explainable AI (xAI) :

In the context of our LSTM model for algorithmic trading, leveraging state-of-the-art explainability techniques like SHAP (SHapley Additive exPlanations), model-agnostic methods such as LIME (Local Interpretable Model-agnostic Explanations), and Partial Dependence Plots (PDP) can provide insights into how the model is making decisions and enhance transparency.

1. SHAP (SHapley Additive exPlanations)

SHAP values provide a way to fairly distribute the contribution of each feature to the prediction across all possible feature combinations. You can use the shap library to compute SHAP values for your LSTM model.



2. LIME (Local Interpretable Model-agnostic Explanations)

LIME creates local surrogate models to approximate the behavior of the complex LSTM model for specific instances. This helps in understanding predictions at an individual level.

3. Partial Dependence Plots (PDP)

PDPs visualize the marginal effect of a feature on the model's predictions while keeping other features constant. Although PDPs are typically used with tree-based models, you can approximate them for your LSTM model.

Conclusion:

In summary, the development of an algorithmic trading model for the BTC/USDT crypto market using LSTM technology holds great promise. The model's architecture, carefully crafted to include trend-following, mean-reversion, momentum-centric, and machine learning strategies, reflects a comprehensive approach to understanding the complexity of cryptocurrency markets. By incorporating historical price data, trading volumes, and various technical indicators, the model seeks to capture patterns, providing a versatile tool for making informed trading decisions.

The flexibility of the LSTM architecture, coupled with strategic design choices such as ensemble methods, dynamic thresholds, and optional reinforcement learning, ensures adaptability to evolving market conditions. Regular fine-tuning, hyperparameter optimization, and rigorous backtesting form integral parts of this approach, reinforcing the model's robustness. While challenges is persist in the volatile cryptocurrency landscape, the proposed model stands as a dynamic solution, poised to navigate market complexities and contribute to a more informed and effective algorithmic trading strategy.